



Two Recommending Strategies to Enhance Online Presence in Personal Learning Environments

Samuel Nowakowski, Ivana Ognjanović, Monique Grandbastien, Jelena Jovanovic, Ramo Šendelj

► To cite this version:

Samuel Nowakowski, Ivana Ognjanović, Monique Grandbastien, Jelena Jovanovic, Ramo Šendelj. Two Recommending Strategies to Enhance Online Presence in Personal Learning Environments. Springer Science+Business Media New York 2014. Recommender Systems for Technology Enhanced Learning: Research Trends and Applications, 2014, Recommender Systems for Technology Enhanced Learning: Research Trends and Applications, 10.1007/978-1-4939-0530-0_11 . hal-01102265

HAL Id: hal-01102265

<https://hal.inria.fr/hal-01102265>

Submitted on 12 Jan 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Two Recommending Strategies to enhance Online Presence in Personal Learning Environments

Samuel Nowakowski¹, Ivana Ognjanović^{2,4}, Monique Grandbastien¹,
Jelena Jovanovic³, Ramo Šendelj^{2,4}

¹ LORIA – Université de Lorraine - Campus Scientifique- BP 239
F-54506-Vandoeuvre les Nancy Cedex – <first name.name@loria.fr>

² Faculty of Information technology, Mediterranean University, Montenegro
– <first name.name@unimediterran.net>

³ FOS-Faculty of Organizational Sciences, University of Belgrade
Jove Ilica 154 - Belgrade, Serbia, jeljov@gmail.com

⁴ Institute of Modern technology, Montenegro

Abstract.

Aiming to facilitate and support online learning practices, TEL researchers and practitioners have been increasingly focused on the design and use of Web-based Personal Learning Environments (PLE). A PLE is a set of services selected and customized by students. Among these services, resource (either digital or human) recommendation is a crucial one. Accordingly, this chapter describes a novel approach to supporting PLEs through recommendation services. The proposed approach makes extensive use of ontologies to formally represent learning context that, among other components, includes students' presence in the online world, i.e., their online presence. This approach has been implemented in and evaluated with the OP4L (Online Presence for Learning) prototype. In this chapter, we expose recommendation strategies devised for OP4L. One is already implemented in OP4L, it is based on the well-known Analytical Hierarchical Process (AHP) method. The other one which has been tested on data coming from the prototype is based on the active user's navigation stream and used a Kalman filter approach.

Keywords. Web-based learning, social presence, online presence, ontology based resource recommendation, Kalman filter, learning trajectories, AHP, CS-AHP

1 Introduction

Web-based Personal Learning Environments (PLEs) have been increasingly adopted by the TEL research community as a mean to support and facilitate online learning practices [1]. From the technical perspective, a PLE is a customizable set of tools and services aimed at

enhancing students' learning experiences and learning outcomes. Among these services, resource (either digital or human) recommendation is a crucial one, given the number and the diversity of available resources on the Web. Various approaches have been proposed to improve the recommendation of resources and adapt them to the learners' needs [3], [52], [53]. They all rely on a learner profile and include a more or less rich description of the learning context, often based on ontologies.

In the last few years, we are witnessing a steady increase in the students' use of Web-based social software tools. This has led to the emergence of novel forms of social presence in online learning environments, PLEs being no exception. Hence, the dominant forms of establishing and maintaining social presence become online status updates, online visibility, availability for online communication and the like. Semantic Web technologies, ontologies in particular, allow for taking these forms of social presence into account when generating recommendations for students.

This chapter reports on the recommendation strategies that have been implemented in the OP4L (Online Presence For Learning) [8] framework and evaluated using the data coming from the students' use of the framework. The first approach is based on the well-known Analytical Hierarchical Process (AHP) method and its adoption for handling conditionally defined preferences, named Conditional Stratified AHP (CS-AHP) method. The *PeerRec* service is developed based on the adoption of prioritization algorithms in a PLE. Based on the students' preferences about some important features of the learning process, the service offers recommendation of peers to communicate and/or collaborate with.

The second approach is based on the active user's navigation stream: we consider that users browsing the Web can be seen as objects moving along trajectories in the Web space. Having this assumption, we derive the appropriate description of the so-called recommender space to propose a mathematical model and state estimate based on a Kalman filter describing the behaviour of the users along the trajectories of the recommender space.

We present the theoretical background, and report on the obtained results and performances. The chapter concludes with perspectives for further developments and prospective evolution studies.

2 Background

In her “vision” paper on the design of social learning environments [2], Vassileva identifies three main roles that should be performed by PLEs: (1) support the learner in finding the right content (right for the context, particular learner, specific purpose of the learner and pedagogically), (2) support learner to connect with the right people (...) and (3) motivate/incentivize people to learn. To devise PLEs with such features, TEL researchers and developers rely on the body of knowledge and experiences originating from several interrelated research domains. The discovery and retrieval of learning resources is one of those domain, and has been widely investigated, beginning with the work on metadata interoperability, then going on with the use of ontologies to better match the learners’ needs and context. As social web applications, such as collaborative tagging, became available, solutions mixing both ontology- and folksonomy-based approaches were proposed. Meanwhile, the recommender systems community developed powerful algorithms for the e-commerce sector, and PLE developers tried to adapt them to e-learning purposes [3], [52], [53].

Social presence is another relevant research field. It has been identified as a crucial success factor in e-learning for many years [4], [5], [6]. At the beginning of e-learning practices, social presence was mostly implemented through online forums and Instant Messaging tools that allowed for establishing and maintaining social presence in online learning settings. The wide adoption of social web applications, such as online social networks, resulted in the inclusion of these applications and connections that students had established in them into online learning environments. Though in theory students can interact with their entire social network, in practice they do not get any indicator about who is really available in the given moment and who is really capable of helping in the current task. Although recommending knowledgeable people for performing a given task is not new, it has been mostly investigated in company settings such as reported, for instance, in [7]. The OP4L framework brings solutions for the two aforementioned challenges in the manner described in the following sections.

3 OP4L framework

3.1 Background and objectives

The OP4L project was running between 2010 and 2012. Its aim was to explore the use of Web-based tools and services for supporting social presence in online learning environments, and thus lead to an improvement in the students' learning experience. In this chapter, we use the term OP4L to name both the project and the developed prototype.

OP4L defines online presence as a temporary description of a user's presence in the online world. It can be considered as an image that a person projects about him/herself into the online world. We explored online presence in the context of DEPTHs [9], a PLE customized for the domain of Software Design Patterns. It makes use of ontologies as a common foundation for the integration of different systems, services and tools in a common environment for collaborative learning of software design patterns. OP4L extends the set of services offered by DEPTHs, by processing online presence data at the semantic level.

3.2 Functional description

A complete technical description of the OP4L framework can be found in [10] and [11], as well as in deliverables available on the project's web site¹. Therefore, in the following we give just a brief overview of the OP4L framework in order to draw an overall picture of the services it offers and the ontologies that make possible the provided services.

From a functional perspective, the primary goal behind the OP4L framework was the development of a *context-aware* PLE through integration of *learning context* data from different learning systems/tools/services, using a flexible *ontology-based* model [14]. We define learning context, i.e., the context of a given learning situation as an interplay of the following main components:

- the learning activity that was performed or the learning-related event that occurred,

¹ <http://op4l.fon.bg.ac.rs/>

- the content that was used and/or produced during the learning activity,
- the individual(s) involved (e.g., learners, teachers, experts) and their respective online presence statuses,
- the (online) environment where the learning activity took place,
- the time when the learning activity took place.

The notion of learning context is formally modeled through an interlinked set of ontologies collectively named LOCO (Learning Object Context Ontologies) framework [9]. Within the OP4L project, the notion of learning context is extended to include the notion of Online Presence. Accordingly, links have been established between the existing LOCO ontologies and the Online Presence Ontology (OPO) [13] to allow for explicitly defining the semantic of this extended notion of learning context. These ontologies served as the foundation for the development of the OP4L prototype with the following main features:

- Integration of data and resources from diverse learning applications that students interact with;
- Context-aware recommendation of resources on software design patterns from online repositories, learning artifacts produced and shared by peers, software projects, discussion threads, chats, etc.;
- Context-aware recommendation of other students, experts and/or teachers to offer help in the given situation.

These services make use of learners' overall learning context, including his/her online presence data, when providing them with recommendations about whom to ask for help or collaborative work. These data are periodically "pulled in" the OP4L system by specific software modules developed for that purpose. Within the online presence data, a key indicator is the "online status" [12] as declared by the user. For instance, a peer whose online status indicates that he/she is busy in the given moment will not be recommended; on the other hand, the system would recommend a face-to-face study session with a peer who has just checked in the same building and whose status indicates that he/she can be freely contacted.

For course designers, one of the main challenges is to adapt interactions to the students' state of presence and to provide services so that interactions can be established smoothly among the participants.

3.3 Main features of OP4L prototype

OP4L services are accessible through a dedicated Moodle platform. The services become available after a student selects a course to study (e.g., the Design Pattern course) and a learning activity (e.g., updating patient's data problem using the UML modeling tool). Fig.1 shows the OP4L online presence services as presented to the student in the user interface. Specifically, they appear in the form of an online presence box in the upper left side of the screen. The box indicates who is competent for the given problem and available online for help or collaboration. It also indicates how to contact potential helper(s)/collaborator(s), either on the Moodle platform itself, or via Facebook or Twitter.

Services linking Moodle to Facebook and Twitter have been developed so that each student can remain using his/her current application, for instance, Moodle for the student looking for help/collaboration and Facebook for the student being contacted. Based on the online statuses declared by the peer students, the learner who is looking for help/collaboration will know in which manner he/she can communicate with the peers. In the case presented on Fig.1, all of the peers can be contacted on Facebook and by email. The system offers several other services as described below.

The screenshot displays the OP4L prototype interface within a Moodle environment. The page title is 'Software engineering - Demo course'. The breadcrumb trail shows 'DEPTHS > S13 > Design Problem > Updating patient's data problem UML modelling tool'. On the left sidebar, there is a 'Recommended peers' section listing three users: Zoran Jeremic, Mira Brkovic, and Kevin Hall, each with a status icon and a 'Configure your account' link. Below this is a 'People' section with a 'Participants' link. The main content area is titled 'Updating patient's data problem UML modelling tool'. It contains a 'Main Project Details' section with a 'Project Description' and a 'Recommended reading' list. Below this is a table showing task progress for three tasks: 'Brainstorming', 'Submitting UML solution', and 'Project assessment'. The table includes columns for 'Task Name', 'Teacher Mark', 'Students' Mark', 'Self Mark', and 'Progress'. The 'Project assessment' task shows a progress of 0/100 (10%). Below the table, it states 'Current Overall Mark = no marks made yet'. At the bottom, there is a 'Legend' section with four color-coded boxes: 'Task has not opened yet' (blue), 'Task is open' (green), 'Task has ended' (yellow), and 'Task late submission closed' (red).

Task Name	Teacher Mark	Students' Mark	Self Mark	Progress
Brainstorming				Done
Submitting UML solution				Not done
Project assessment	0/100 (60%)	0/100 (30%)	0/100 (10%)	Not done

Current Overall Mark = no marks made yet

Legend

- Task has not opened yet
- Task is open
- Task has ended
- Task late submission closed

Fig. 1. Inside the prototype with recommended peers on the left

The system recommends appropriate contents related to the topic of the course (Fig. 2). Its originality is to augment the course digital library with resources brought in and built by the students during the course.

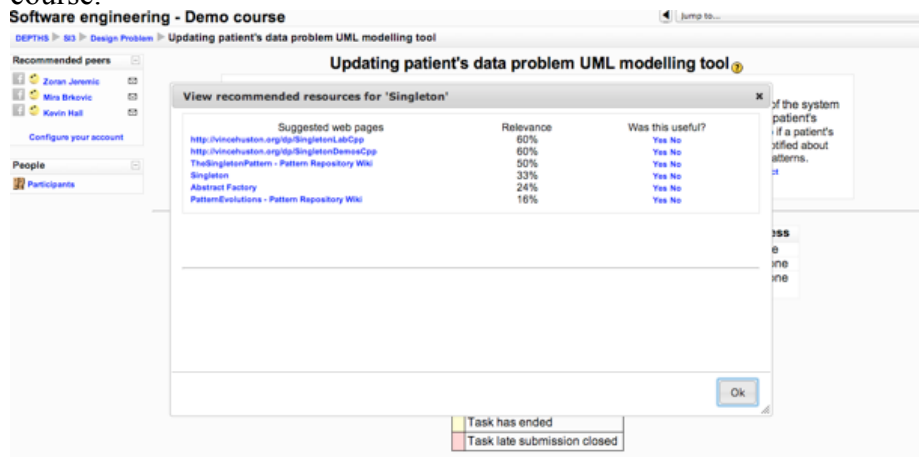


Fig. 2. Interface illustrating recommendation of digital resources

For enhancing collaboration, students are also given a brainstorming tool where ideas can be annotated and rated. Finally, students can upload their work and benefit from peer evaluations, as shown in Fig. 3. They can assess solutions proposed by fellow students only after uploading their own solution.



Fig. 3. Interface illustrating evaluation of other students' work.

4 Recommendations strategies

To support students through recommendation of digital and human resources, the OP4L framework needs to implement recommendation algorithms. In this section, we present the algorithm that has been implemented to recommend peers and digital content. We also present a complementary approach based on the students' learning trajectories.

4.1 Recommendation algorithm based on the Conditional Stratified Analytic Hierarchy Process

In a learning environment each individual student has different characteristics, motivation and performance, which all together indubitably should be considered when designing and/or adapting the learning process. It has been well recognized that there is a need to move away from the 'one size fits all' paradigm, and to offer personalized learning experience to learners [58]. Most of the approaches aimed at adapting the learning process to individual learners or learning groups have been based on the learners' level of knowledge [59], [60]. Other learner features taken into account are background, hyperspace experience [61], preferences and interests, as well as learning styles and their effect on learning achievements [62]. However, despite interests in exploring these diverse learners' features, in recent years, modeling of adaptive systems has still revolved around acquiring and representing learners' knowledge. This orientation does not properly reflect real life situations where each learner has a variety of selection criteria and requirements over them when choosing other learners for collaboration and cooperation [39].

Representation and analysis of preferences have been studied in many fields such as economics, especially in project and risk management, decision theory, social choice theory, with further developments and applications in areas such as operational research, databases, security analysis, and artificial intelligence [37]. Modeling of user preferences is a great challenge, as it is difficult to express human opinion in a way that can be easily processed by computers [43]. The adoption of user preferences in the design of a PLE brings in additional challenges in coordination of the flow of information among the learners involved in a learning process, and in encouragement of interactions across learning systems/tools/services. In particular,

communication between peers may be induced by different needs and expectations (e.g., general questions about the course organization, help needed in learning and understanding of some course topics, etc.). In some cases, urgent response is needed, while in others, only answers of good learners or good senior learners are useful. The following problems emerge from this observation: (i) characterization of peers for communication related to the selection criteria, and (ii) definition of specific requirements and preferences over them. Furthermore, it is reasonable to expect that appropriate peers are not available all the time for each learning topic, or the most appropriate peers are not there when a specific learner needs their assistance. Having all this in mind, the *PeerRec* service, developed as a part of the OP4L framework, integrates: i) the aforementioned semantic representation of online presence data (see Section 3.2); ii) two-level hierarchical structure of concerns and qualifier tags for semantic representation of selection criteria proposed in [46]; and iii) the CS-AHP prioritization algorithm for presentation and ranking of users' preferences [46].

4.1.1. CS-AHP algorithm

CS-AHP (Conditional Stratified Analytic Hierarchy Process) adopts Analytical Hierarchy Process (AHP) technique [42] for different kinds of preferences using a two-level hierarchical structure of concerns and qualifier tags. Concerns are a set of quality characteristics that represent important matters of interest for learners such as fields of professional specialization, spoken languages or preferred message response time. Qualifier tags represent possible values for each concern (e.g., qualifier tags for spoken languages could be *goodLevel*, *mediumLevel*, *lowLevel*, *unknownLanguage*).

Following the well-known AHP framework for expressing and ranking user requirements, the *PeerRec* service enables learners to express their requirements by defining relative importance between concerns, and between qualifier tags of each concern. Relative importance is typically defined with odd numbers ranging from 1 (equal importance) to 9 (extreme importance of one concern over the other). The options available to learners in respect to their online presence (i.e. peers for collaboration) are also associated with qualifier tags. Once the relative importance is set between all pairs of concerns, the AHP algorithm performs a tuned pair-wise comparison of the

learners' requirements. The outcome of this process are ranks $\{r_1, \dots, r_n\}$, which provide values from the $[0,1]$ interval over the set of available options. The process is done in two main steps: i) the set of concerns and their qualifier tags are locally ranked with ii) rank of each available option is calculated based on the ranks of the qualifier tags associated with that option.

CS-AHP also allows for setting conditional preferences. For example, learners are often aware that requirement of expertise is hard to meet, so they may define a compromise: they are only prepared to wait for response from a learner with expert knowledge, if the waiting time is kept at a very minimum; otherwise, they are willing to contact learners with lower level of expertise but with medium delays in response. CS-AHP is simple to perform, and requires quadratic number of comparisons, which brings linear time complexity to the number of available options [46].

4.1.2. Applying the CS-AHP Algorithm in a PLE

In order to develop a user-friendly service for the implementation of CS-AHP in a PLE, a structure of concerns and qualifier tags is built on the Preview framework related to learning environments [39]. The following three categories of concerns are recognized from the aspect of definition and management issues [38]: statically defined concerns, concerns dynamically defined for each conversation, and concerns with dynamic updates.

Statically defined concerns present general information about a learner who is looking for another peer appropriate for conversation; examples include the known language(s), preferred subject area, and availability for F2F contact. The recognized concerns are defined as attributes in the learner's profile developed for the *PeerRec* service. Information in the profile may be changed upon a learner's request; otherwise, it remains unchanged, representing the learner's characteristics that are used in each call of the service.

On the other side, some concerns are directly related to the context of the upcoming conversation (i.e. concerns dynamically defined for each conversation), and, thus learners are enabled to define them explicitly. For example, a learner may define the type of conversation (e.g. the help in understanding, etc.) and/or the urgency level (e.g. extra urgent) (see Fig. 4).

Tom ↔ Sue

[Remove contact](#) | [Block contact](#)

[All messages](#) | [Recent messages](#)

No messages were found

Course

Urgence ☐ Not urgent ☐ Next three days ☒ Next day ☐ Next twelve hours ☐ Next three hours

Request type ☒ Learning help ☐ Administration/service question ☐ Other



Message

You are logged in as [Tom](#) ([Logout](#))

Fig. 4. Definition of the *Urgence level* and *Request Type* at the beginning of each conversation

Contrary to the concerns defined in the learner's profile (with qualifier tags statically defined and changeable only on the learner's explicit request), concern defined as *ConversationRate* should be updated for each completed conversation. To this end, the PeerRec service asks learner to rate each peer after communication (see button *Rate conversation* on Fig. 5), and based on the learner's feedback updates the values to be used in further communications with the same peer (see attributes *Response speed* and *Response relevancy* on Fig. 6). In cases when learner is not interested in setting rates (from different personal reasons ranging from current disengagement, lack of time or interests), he/she may select the N/A option meaning that experience from the latest conversation does not bring any change to the

aggregated rates. Furthermore, if no rate is previously aggregated, initial selection of N/A represents indifferent and undeclared learner (neither positive nor negative rate is specified).

Tom ↔ Sue
[Remove contact](#) | [Block contact](#)

[All messages](#) | [Recent messages](#)

Friday, 22 June 2012

Course: Databases I
 Urgence: Next day (due 23 June 2012 at 12:34 PM)
 Request type: Learning help

12:34 PM: Dear Sue,
 Can you suggest some literature about differences between Oracle 10g and 11g edition. It would really help for my paper due next week.

18:47 PM: Of course.
 For start look at http://www.dba-oracle.com/oracle11g/oracle_11g_new_features.htm and <https://forums.oracle.com/forums/thread.jspa?threadID=646818>

Course

Urgence ☐ Not urgent ☐ Next three days ☐ Next day ☐ Next twelve hours ☐ Next three hours

Request type ☐ Learning help ☐ Administration/service question ☐ Other

Message

Response speed ☒ In time, very satisfactory ☐ In time, satisfactory ☐ Over due, but satisfactory ☐ Over due, dissatisfactory

Response relevancy ☐ Very relevant ☒ Relevant ☐ Somewhat relevant ☐ Not relevant

You are logged in as [Tom](#) ([Logout](#))

Fig. 5. Rating conversations based on *Response speed* and *Response relevancy*

Tom ↔ Sue

[Remove contact](#) | [Block contact](#)

Previous ratings:
Response speed: 4/4
Response relevancy: 3/4

[All messages](#) | [Recent messages](#)

(No messages were found)

Course

Urgence ☐ Not urgent ☐ Next three days ☐ Next day ☐ Next twelve hours ☐ Next three hours

Request type ☐ Learning help ☐ Administration/service question ☐ Other

Message

You are logged in as [Tom](#) ([Logout](#))

Fig. 6. Aggregated conversation ratings (*Response speed & Response relevancy*)

After invoking the PeerRec service, all currently available peers are ranked in decreasing order (based on the results obtained from applying the CS-AHP algorithm), and the learner can decide with whom to start a conversation (see Fig. 7).

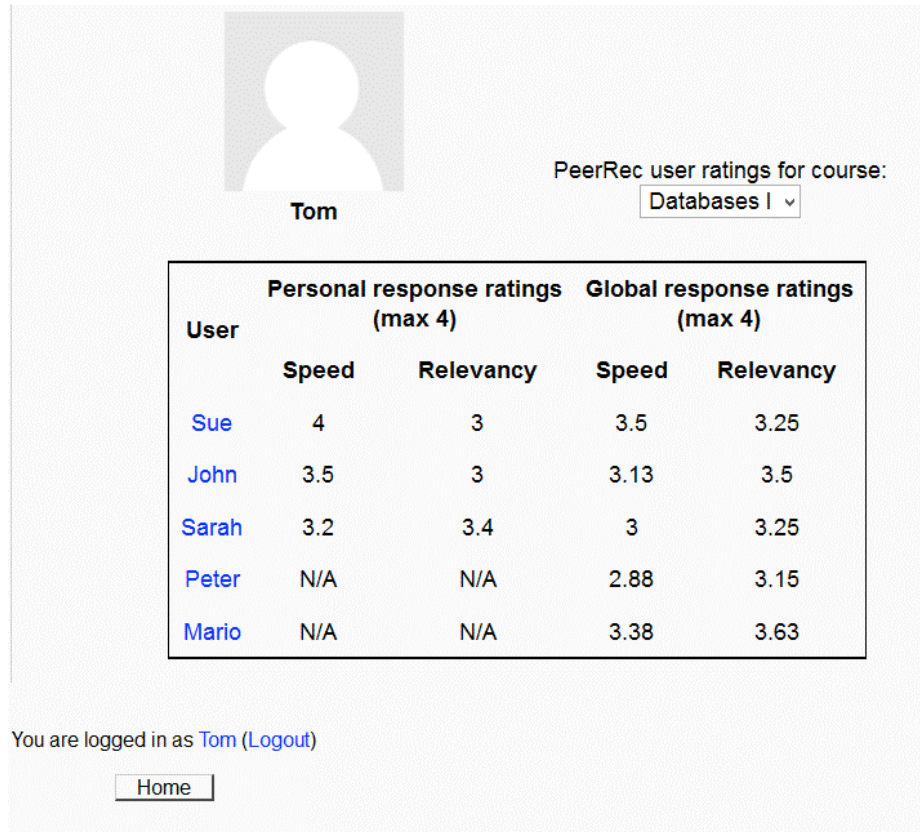


Fig. 7. The *PeerRec* service invocation and ranking of all peers available for communication

5 Discussion and Related Work

Previous analyses from [46] give a frame for addressing different preference structures with scales of input and output information with different semantics [40], [41], [46], [54], [48], [55], [47]. Unique representational and reasoning technique that can effectively address different kinds of preferences and reasoning queries does not exist. Therefore, selection of the most appropriate method should be done depending on the characteristics of the problem at hand in the given field [48].

In the learning environment, different aspects of recommendations are considered and analyzed, according to their own mission, vision, and objectives [63],[64],[65],[67] such as, dealing with the assessment of the student's learning performance, providing course adaptation and

learning recommendations based on the student's learning behaviour, evaluation of learning material and educational web-based courses, developments for detection of atypical students' learning behaviours, etc. Some of them are focused on creation of Recommendation Systems that link users with items (course, learning material, peers for communication,...) [66], associating the content of the recommended item or opinion of other individuals with the actions or opinions of the original users of the system.

However, to date, there has been comparatively less progress in direction of focusing to students' requirements and preferences [67], although there is currently an increasing interest in applying techniques and methods primary developed and used in other fields to the educational environment [68]. In this context, recently developed CS-AHP algorithm extends well-known Analytical Hierarchical Process (AHP) proposed by Saaty [42] which is a widely adopted multi-criteria decision making method to make complex decisions [44],[45]. Also, the use of two-layered structure of concerns and qualifier tags has several explanations, sufficient expressiveness (according to [43]) and analogous to the concept of attributes in all developed techniques for addressing unconditional and conditional preferences [57].

6 Learning trajectories algorithm

In this section we present an approach that does not limit itself to the content recommendation, but aims at analyzing the way that students learn. This approach starts from the following hypothesis: learning is linked to the identity; so, when digital learning environments are used, we consider that digital identities have to be studied.

From a philosophical point of view, digital identities can be seen as the "sèmes" which identify an individual in its singularity. We can then speak of "individuation" as a process leading from the undifferentiated to uniquely defined and personalized. Consequently, we can identify someone by his/her personal trajectory in the cyberspace. To validate this approach, we have implemented a service (using Matlab) with input data from the OP4L learning framework. The following sections show the theoretical background of our approach, as well as some numerical results and perspectives.

6.1 Input Data

The input data we consider is made up of traces that users have left (log files and users' ratings) when interacting with the system (intranet, Web site, etc.). These data are first used to discover patterns of usage, and then to perform recommendations. In our case, data are coming from the log files of the OP4L prototypes. They show (for each user) the pages visited in the learning environment.

6.2 Markov model

One well-known approach to making use of users' online history is to compute predictions by using Markov models. The use of Markov models in the frame of the Web has been first dedicated to the reduction of access time by pre-fetching and caching pages [30]. With the same goal, Box and Jenkins in [20] estimated conditional probabilities of transitioning directly from one page to another within a given time. First order Markov models are not very accurate in predicting the user's browsing behavior since these models do not look far in the past to efficiently discriminate different histories [21]. Pirolli and Pitkow in [32] and [33] showed that the prediction accuracy is increased when using a longer history. Higher order Markov models, also called k^{th} order Markov models, are used to capture longer histories. Given the navigation history of size k , the probability of each resource is computed, and the resources with the highest conditional probability are recommended. The use of k^{th} order Markov models lead to a high accuracy.

Let us notice that k^{th} order Markov models are similar to frequent contiguous patterns of fixed size $k + 1$ in the case when support and confidence thresholds are set to 0. One drawback of k^{th} order Markov models is the storage requirements; indeed, in a k^{th} order Markov model a huge number of states are handled (this number increases according to the order of the model) [30]. Moreover, as with previous approaches, we are faced to a reduced coverage due to the problem of matching the active history and training data. Many approaches can overcome coverage limitation. For example, we can mention the development of Markov models of orders varying from 1 to k called the all k^{th} order Markov model [29]. However such a model dramatically increases the complexity and storage space drawbacks.

When using Markov models, the order of navigation is taken into account and the sequences are strictly contiguous, hence, these models are not permissive. If a given user performs parallel navigations or goes to an unwanted resource (noise), the model cannot correctly handle such a behaviour and will thus reduce the size of the history considered. Such situations are handled by association rules and sequential patterns as resources are not contiguous. Moreover, when the model does not match the complete history, the most distant consulted resources are discarded for computing predictions. Thus, the most recent resources are always considered while some of them may be not important or may be navigation mistakes and should be discarded.

In the following section, we show how we can derive from Markov model an approach based on Kalman filtering and target tracking. In our approach, we based our recommendation strategy on a transformation of the web space.

6.3 Principles

Kalman filter is an optimal state estimator of a linear system [16]. It can estimate the state of the system using a priori knowledge of the evolution of the state and the measurements. Kalman filter has main applications in control systems and target tracking.

6.3.1. Target tracking in the cyberspace

We consider a user who browses Web pages or online resources. Each page/resource belongs to a category (categories are related to the classification of the available resources). We then consider that one category corresponds to one dimension of a space. And the aggregation of all the dimension builds a space having as dimensions as there are categories of resources. Then, all possible categories define the geometrical structure of the space.

Knowing that to see one resource corresponds to a specific position in the space i.e., a specific vector. Successive vectors (\vec{v}_1 to \vec{v}_n) give successive “positions” in the space. In the context of online learning, these “positions – learning positions” define the trajectory of a user in the recommender space.

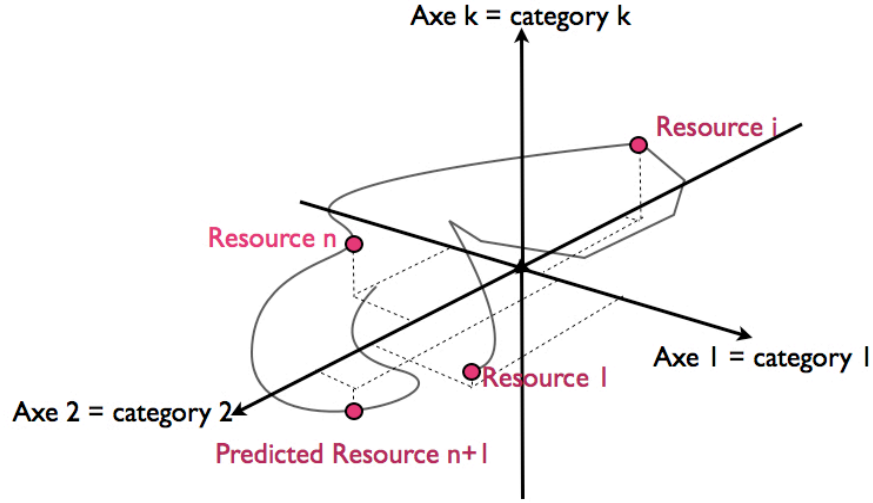


Fig. 8. Learning trajectory

Each vector has the following dimensions:

$m \times 1$, m is the number of categories; each seen page belongs to one or more categories. In our case, i.e., the analysis of the learning processes on the OP4L platform, each page coming from the tested site is classified (courses, exercises, ...). The structure of each vector will be as follows. Considering that we have m possible categories for the resources, to see one specific resource is to have a vector containing “1” in the corresponding row, and “0” elsewhere.

$$\begin{bmatrix} cat-1 \\ cat-2 \\ \dots \\ cat-i \\ \dots \\ cat-m \end{bmatrix} \leftarrow \text{corresponding category} \quad \begin{bmatrix} 0 \\ 0 \\ \dots \\ 1 \\ \dots \\ 0 \end{bmatrix} = \vec{v}_k \quad (1)$$

6.3.2. Kalman filter: equations

Hypothesis

Our main hypothesis is the following: considering that users are moving along a trajectory defined by a set of vectors, we assume that the user can be considered as a target which is described by three components in the state space, i.e., position, speed and acceleration. These three components will completely describe the dynamics of the

moving users [16], [20], [23]. Thus, we choose to represent the state vector by concatenating these three components. The state vector has the following form [51]:

$$X_k = \begin{bmatrix} x \\ v \\ \gamma \end{bmatrix}_k \quad (2)$$

where: $\dim(X_k) = 3m \times 1$

- x contains the components of the position vector, dimensions $m \times 1$
- v contains the components of the speed vector, dimensions $m \times 1$
- γ contains the components of the acceleration vector, dimensions $m \times 1$.

The dynamic of this state vector is modeled by a state space model of the following form:

$$\begin{cases} X_{k+1} = AX_k + w_k \\ Z_k = HX_k + v_k \end{cases} \quad (3)$$

Matrix A includes the relationship between the position, its first and second derivations will inform us on the geometrical characteristics of the trajectory. This is the matrix form of the cinematic equation linking position to speed and acceleration. T is a parameter that introduces time in the equation. In our case, we consider T equal to 1 because time is fixed each time the user goes to another webpage. The results of the algorithm are not sensitive to T .

$$A = \begin{bmatrix} \alpha & T & \frac{1}{2}T^2 \\ 0 & \alpha & T \\ 0 & 0 & \alpha \end{bmatrix} \quad (4)$$

Where: $\dim(A) = 3m \times 3m$

Many values of parameter α have been tested. The chosen value does not influence our numerical results.

w_k and v_k (equation (3)) are random noises (their properties will be given in the next section) which take into account unexpected variations in the trajectories.

Matrix H (equation (3)), called the measurement matrix, is structured to obtain the values of the positions in the recommender space. Thus, H will have the following structure:

$$H = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ \dots & \dots & 1 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \end{bmatrix} \quad (5)$$

Where: $\dim(H) = m \times 3m$

6.3.3. General equations of the filter

Having the state space model (equations 3) and the structure of the state vector, we can derive the equations of the filter. First, we present some important properties of the Kalman filter:

- Information about X and Z is given as a Markov model i.e., Z is a linear combination of the components of X;
- Estimations of X are obtained from any initial instant;
- Estimations can be obtained for non-stationary process i.e., time-varying models.
- w_k and v_k are uncorrelated white noises where $w_k \approx N(0, Q)$ and $v_k \approx N(0, R)$.

The Kalman filter equations comprise the following equations [8]:

Prediction: it is the predicted state knowing past values:

$$\begin{cases} \hat{X}_{k+1/k} = \hat{X}_{k/k-1} + K_k (Z_k - H\hat{X}_{k/k-1}) \\ \quad = (A - K_k H) \hat{X}_{k/k-1} + K_k Z_k \end{cases} \quad (6)$$

Kalman gain: it describes the dynamic of the filter. The dynamic takes into account the variations of the moving target.

$$K_k = AP_{k/k-1}H^T (HP_{k/k-1}H^T + R)^{-1} \quad (7)$$

The evolution of the uncertainty on the estimation is then given by the following Riccati equation:

$$P_{k+1/k} = AP_{k/k-1}A^T - AP_{k/k-1}H^T (HP_{k/k-1}H^T + R)^{-1} HP_{k/k-1}A^T \quad (8)$$

where the initial conditions (which initialize the filter) are given by:

$$\hat{X}_{0/-1} = X_0, \quad P_{0/-1} = P_0 \quad (9)$$

and the state prediction is given by: $\hat{X}_{k+1/k}$

The Kalman predictor will predict the future position in the recommender space i.e. the most possible category knowing the past of the user.

6.3.4. Recommendation strategy

The user profile is built from the list of pages visited on the platform. Each page/resource is defined by a subset of categories such as “modelling”, “courses”, etc.

Our new recommending strategy is based on the control loop shown in Fig. 9.

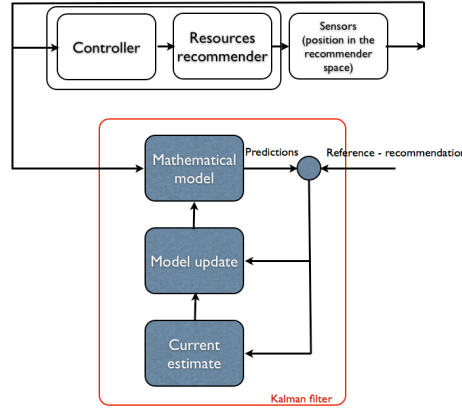


Fig. 9: Control loop for recommendation

This control loop will observe the difference between the estimated value of the category and the calculated category, and will integrate the controller/recommender to build the most accurate model of the user. Hence this configuration can predict where the user will “move” in the recommender space. The recommendation strategy will use the predicted position to “suggest” to the user the appropriate category of content.

Conversely to existing methods that recommend specific content items to a given user, this method performs on the macroscopic level, i.e., subspaces of specific categories. The strategy isolates the appropriate subspace and the recommendation is done in the related categories. Then, we can imagine providing a more precise

recommendation by doing the second iteration of computing on the subspace (target tracking in the trajectory in the subspace and positions prediction) – a kind of zoom effect. To summarize, the recommendation is based on two arguments:

- the user's actual state of mind
- a subset of retained dimensions

We then have a set of items to be recommended. Furthermore, according to the pages the user has visited during the day, we can refine our recommendation.

6.4 Results

In order to model the trajectories, we have to identify categories of pedagogical resources that define the recommender space. Based on the extensive log analysis, we propose the following basic description:

- assessment (MA)
- courses (C)
- resources (R)
- forum (F)
- modelling workgroup (MW)
- modelling brainstorm (MB)
- modelling (M)

Each position vector is built using relation (1). We obtain position vectors containing only “0” and “1” on the appropriate dimension.

Thus, we consider that users are moving in a 7 dimensions space. Using our approach and algorithms, we obtain the following results for a specific user (Fig. 10 and Fig. 11). In the following figures, we show one learner’s trajectory along specific dimensions of the space (Dimensions MA, C and R). In both figures, X-axis represents time and Y-axis represents the seen item thus it is “0” or “1”.

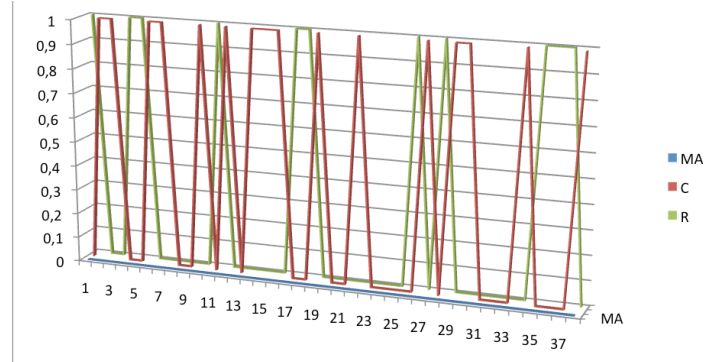


Fig. 10: Evolution in MA, C and R dimensions

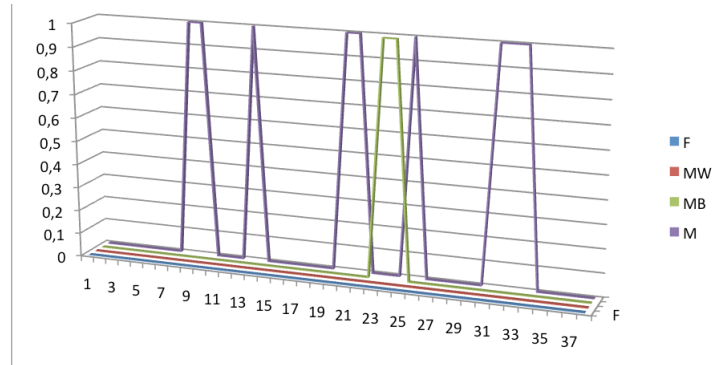


Fig. 11: Evolution in F, MW, MB and M dimensions

Combining all these evolutions, we obtain trajectories in 7-dimensions space. Applying our tracking algorithm, we can compute the next viewed category of contents. For example, we can show the results in one dimension, dimension R (Fig. 12).

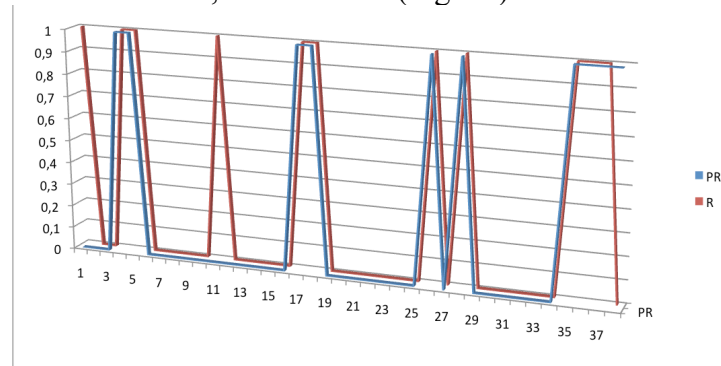


Fig. 12: Comparison between Real Dimension R and predicted one PR

Figures 10 and 11 show the evolution of the trajectory along the dimensions of the recommender space. Figure 12 shows the predicted trajectory to the real trajectory relatively to one dimension of the recommender space – dimension R. We can see that the prediction PR follows the real trajectory R. Having this prediction (PR), we can now derive a specific recommender system based on these categories. For example, using the information given by the prediction, the system can recommend specific content items to the student.

Thus, the strength of our approach is in its capability to make recommendations that consider the users' habits, i.e. give the main directions to follow knowing the trajectory in the space and not to suggest specific resources.

7 Conclusions and perspectives

The OP4L prototype implementing the CS-AHP algorithm has been used with several groups of students in France, Slovenia, Macedonia and Serbia during the academic year 2011-2012. It was mainly evaluated with undergraduate students in Computer Science, within a course on Software Engineering. The first results are described in [50], and the students' responses to the questionnaires used in the study do not mention any mistake in the recommended people. The CS-AHP algorithm is working.

The Kalman filter approach differs from other recommendation approaches. Indeed, it focuses on users' behaviour (modeling their path in the space of the resources) to predict the categories of resources that are likely to fit their needs. In this formalism, categories of resources correspond to a subspace of the space of reference, which allows us to identify a set of resources that can meet the needs of the user. In the presented examples, we show that these trajectories can serve as monitoring tools for pedagogues.

We are thus able to describe behaviours by learning trajectories and predict what types of resources a learner is likely to access.

We would like to conclude with the perspectives offered by the learning trajectories algorithm. In particular, the approach based on learning trajectories allows one to:

- model learning process, i.e. to understand how someone learns

- recommend content items more accurately, i.e. knowing about someone's learning process we can recommend exercises, courses, self-assessment, ...
- develop a recommender system based on different levels of analysis, i.e. to identify the accurate subspace and then re-compute in a zoom effect to identify precisely the concerned dimensions.

Moreover, this geometrical approach opens a new field of research focused on the geometrical description of the recommender space, and how this geometry could lead to better recommendation and dynamics understanding. In the OP4L context, this approach of learning trajectory analysis will enable a better understanding of all the experiments done or to be done with students. Thus, the results obtained in the studies done in France, Macedonia, Serbia and Slovenia could be enforced and enriched by learning trajectories modelling. Moreover, this combined methodology will help us to identify the appropriate functionalities, those which are important, those which are not used, and will lead our research on the integration of others social networks (technically and also by recommending to students, some resources coming from the social network), and the definition of new ontologies related to other disciplines.

Acknowledgements:

This work was supported by the SEE-ERA Net Plus program, contract n° 115, from the European Union.

8 References

1. Atwell, G., Personal Learning Environments – the future of eLearning? eLearning Papers, www.elearningpapers.eu, 2 (1)-January 2007 – ISSN, 1887-1542, (2007)
2. Vassileva, J., Towards Social Learning Environments, IEEE TLT, 1 (4), (2008)
3. Vuokari, R., Manouselis, N., Duval, E., eds., Special Issue on Social Information Retrieval for Technology Enhanced Learning, Journal of Digital Information, 10 (2), (2009)
4. Aragon, S. R., Creating social presence in online environments. New Directions for Adult and Continuing Education, 100, 57-68, (2003)
5. Cob, S. C., Social Presence and Online Learning: A current view from a Research Perspective, Journal of Interactive Online Learning, 8 (3), (2009)
6. Lowenthal, P.R., Social Presence, In S. Dasgupta (Ed.), Social computing: Concepts, methodologies, tools, and applications, 129-136, Hershey, PA: IGI Global, (2010)
7. Beham, G., Kump, B., Ley T., Lindstaed, S. N., Recommending knowledgeable people in a work-integrated learning system, 1st RecSysTEL workshop, Procedia Computer Science 1 (2), 2783-2792, Elsevier, (2010)

8. OP4L project's website: <http://op4l.fon.bg.ac.rs/>
9. Jovanovic, J., Knight, C., Gasevic, D., Richards, G. Ontologies for Effective Use of Context in e-Learning Settings. *Educational Technology & Society*, 10(3), 47-59, (2007)
10. Jeremic, Z., Milikic, N., Jovanovic, J., Radulovic, R., Brkovic, M., Devedzic, V., 2011, OP4L : Online Presence Enabled Personal Learning Environments, IEEE – ERK'2011 conference, Portoroz, Slovenia, (2011)
11. Milikic, N., Radulovic, R., Devedzic, V., Infrastructure for Exchanging Online Presence Data in Learning Applications, IEEE – ERK'2011 conference, Portoroz, Slovenia, (2011)
12. Stankevic, M., Modeling Online Presence, In: Proceedings of the First Social Data on the Web Workshop, Karlsruhe, Germany, October 27th, 2008, CEUR Workshop Proceedings, ISSN 1613-0073, online CEUR-WS.org/Vol-405/paper1.pdf, (2008)
13. Jovanović, J., Gašević, D., Stanković, M., Jeremić, Z., Siadat, M., Online Presence in Adaptive Learning on the Social Semantic Web, In Proceedings of the 1st IEEE International Conference on Social Computing - Workshops (Workshop on Social Computing in Education), Vancouver, BC, Canada, pp. 891-896, (2009)
14. OP4L D3.1, OP4L Models, <http://op4l.fon.bg.ac.rs/sites/default/files/OP4LD3.1.pdf>
15. Brooks C., Bateman S., Greer J., McCalla G., Lessons Learned using Social and Semantic Web Technologies for E-Learning in Semantic Web Technologies for e-Learning, Dicheva D., Mizoguchi R., Greer J., eds., IOS Press, (2009).
16. Anderson B., Moore J.B. Optimal filtering. Prentice Hall – Information and System Sciences Series, (1977)
17. BARB: Broadcaster Audience Research Board, DOI=<http://www.barb.co.uk/>.
18. Bernier, C., Brun, A., Aghasaryan, A., Bouzid, M., Picault, J., Senot C. and Boyer A., Topology of communities for the collaborative recommendations to groups. In SIIE 2010, (2010).
19. Bestavros, A., Using Speculation to reduce Server Load and Service Time on the WWW. In Conference on Information and Knowledge Management, pp. 403–410, (1995)
20. Box G.E.P., Jenkins G.M., Time series analysis: forecasting and control. Holden Day, (1970).
21. Deshpande M., Karypis G. Selective Markov Models for Predicting Web Pages Accesses. *ACM Trans. Internet Technol.*, pp. 163-184, (2004)
22. Das A., M. Datar, A. Garg and S. Rajaram , Google News personalization: scalable online collaborative filtering. In WWW'07 Proc. of the 16th International Conference on World Wide Web, pp. 271–280, (2007)
23. Gevers, M., Vandendorpe, L. Processus stochastiques, estimation et prediction. DOI=<http://www.tele.ucl.ac.be/EDU/INMA2731/>, (2011)
24. Gibson W. ,Neuromancien, Collection J'ai Lu, (1988)
25. Goldberg D., Nichols D., Oki B., Terry D. ,Using Collaborative filtering to Weave an Information Tapestry” Communication of the ACM, pp. 61–70, (1992)
26. Goldberg, D., Nichols, D., Oki B., Terry D. Using collaborative filtering to weave an information tapestry,” Communications of ACM, pp. 61–70, (1992)
27. Goldberg D., Nichols D., Oki B., D. Terry D. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, pp 61–70, (1992).
28. Grear M., Mladenic, D., Fortuna, D., Groblenik, M., Data Sparsity Issues in the Collaborative Filtering Framework. *Advances in Web Mining and Web Usage Analysis*, 58–76, LNCS vol. 4811, (2006)
29. Nakagawa N., Mobasher B. Impact of Site Characteristics on Recommendation Models based On association Rules and sequential patterns. In Proceedings of the IJCAI'03 Workshop on Intelligent Techniques for Web Personalization, (2003)

30. Padmanabhan V., Mogul J. Using predictive prefetching to improve World Wide Web Latency. *Computer Communication Review*, pp. 22–36, (1996)
31. Pazzani, M., Billsus D. Content-Based Recommendation Systems. *The Adaptive Web*, pp. 325–341, (2007)
32. Pirolli P., Pitkow J. Distribution of Surfer's paths through the World Wide Web: Empirical Characterizations. In *World Wide Web Journal*, 2 (1-2), pp. 29–45, (1999).
33. Pitkow J., Pirolli P. Mining longest Repeating Subsequences to predict World Wide Web surfing. In *Proceedings Of the 2nd conference of USENIX Symposium on Internet Technologies and Systems*, pp. 139-150, (1999)
34. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Analysis of recommender algorithms for e-Commerce. In *Proceedings Of the 2nd ACM e-Commerce conference*, (2000)
35. Söderström T., *Discrete-time stochastic systems: estimation and control*. Prentice Hall International, (1994).
36. Ungar L., Foster D. Clustering methods for collaborative filtering. In *Proceedings Of the AAI98 Workshop on Recommendation Systems*, (1998),
37. Ognjanović, I., Gašević, D., Bagheri, E., & Asadi, M. Conditional Preferences in Software Stakeholders' Judgments. In *Proceedings of the 26th Annual ACM Symposium on Applied Computing*, pp. 683-690, Taichang, Taiwan, (2011).
38. Ognjanović, I., Šendelj, R. Making judgments and decisions about relevant learning resources, In *Proceedings of the 20th International Electrotechnical and Computer Science Conference, Portoroz, Slovenia (ERK 2011)*, pp.409-412, (2011).
39. Ognjanović, I., Šendelj, R. , Teachers' requirements in dynamically adaptive e-learning systems. In *Proceedings of 4th International Conference on Education and new Learning Technologies (EDULEARN12)*, Barcelona, Spain, (2012).
40. Boutilier, C., Brafman, R. I., Domshlak, C., Hoos, H. H., & Poole, D. ,CP-nets: a tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of AI Research*, 21(1), 135-191, (2004).
41. Brafman, R. I., Domshlak, C., Introducing variable importance tradeoffs into CP-nets. In *The Proceedings of the Eighteenth Conference on Uncertainty in AI*, pp. 69-76, Canada, (2002).
42. Saaty, T. L. ,*The Analytic Hierarchy Process*. McGraw-Hill, New York, (1980).
43. Yu, Z., Yu, Z., Zhou, X., Nakamu, Y. Toward an Understanding of User-Defined Conditional Preferences., In *Proceedings of the 8th IEEE International Conference on Dependable, Autonomic and Secure Computing*, pp. 203-208, Washington, USA, (2009).
44. Büyüközkan, G., Çifçi, G., Güleriyüz, S. Strategic analysis of healthcare service quality using fuzzy AHP methodology, *Expert Systems with Applications*, 38(8), 9407–9424, (2011).
45. Chen, M. K., Wang, S., The critical factors of success for information service industry in developing international market: Using analytic hierarchy process (AHP) approach. *Expert Systems with Applications*, 37(1), 694–704, (2010).
46. Ognjanović, I., Gašević, D., & Bagheri, E. A Stratified Framework for Handling Conditional Preferences: an Extension of the Analytic Hierarchy Process", *Expert Systems with Applications*, 40 (4), pp.1094-1115, (2013)
47. Wilson, N. Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence*, 175(7-8), 1053-1091, (2011).
48. Zavaskas, E. K., Kaklauskas, A., Peldschus, F., & Turskis, Z., Multi-attribute assessment of road design solutions by using the COPRAS Method. *The Baltic Journal of Road and Bridge Engineering*, 2(4), 195-203, (2007).

49. Chen, S., Buffett, S., & Fleming, M. W. Reasoning with Conditional Preferences across Attributes. In Proceedings of the 20th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in AI, (pp. 369-380). Montreal, Canada, (2007).
50. Grandbastien, M., Loskovska, S., Nowakowski, S. and Jovanovic, J. Using online presence data for recommending human resources in the OP4L project. Conference RecSysTel, September 2012, Sarrebrück, Germany, (2012)
51. Nowakowski, S., Boyer, A., Bernier, C. Automatic tracking and control for web recommendation. New approaches for web recommendation. Conference SOTICS 2011, October 2011, Barcelona, Spain, (2011)
52. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R., Recommender Systems in Technology Enhanced Learning. Handbook of Recommender Systems. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds), 387-415, (2010)
53. Santos, O.C., Boticario, J.G., Educational Recommender Systems and Technologies. Practices and Challenges. IGI Global, (2012).
54. Berander, P., & Andrews, A. Requirements Prioritization. Engineering and Managing Software Requirements, 69-94, (2006).
55. Chen, S., Buffett, S., Fleming, M. W. , Reasoning with Conditional Preferences across Attributes. In Proceedings of the 20th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in AI, pp. 369-380, Montreal, Canada, (2007).
56. Behr, M. J., Guershon, H., Thomas, P., Richard, L., Rational number ratio and proportion. In D. A. Grouws, Handbook of Research on Mathematics Teaching and Learning, pp. 296-333, New York: Macmillan, (1992).
57. Ognjanovic I., Gasevic D., Bagheri E., Asadi M., Conditional Preferences in Software Stakeholders' Judgments, *In Proceedings of the 26th Annual ACM Symposium on Applied Computing (SAC 2011)*, Tunghai University, TaiChung, Taiwan, 2011
58. Dagger, D., Wade, V., Conlan, O., Personalisation for All: Making Adaptive Course Composition Easy. Educational Technology & Society, 8 (3), 9-25, (2005).
59. Popescu E., Trigano P., Badica C. , Adaptive educational hypermedia systems: A focus on learning styles. In Proc. of the International Conference on Computer as a Tool (EUROCON), Warsaw, Poland, IEEE Computer Society, (2007).
60. Stash N, De Bra P., Incorporating cognitive styles in AHA! (The Adaptive Hypermedia Architecture). Proceedings of the International Conference Web-Based Education (IASTED), Innsbruck, Austria, 378-383, (2004).
61. Brusilovsky P. , Adaptive hypermedia. User modeling and user adapted interaction. Tenth year anniversary issue (Alfred Kobsa, ed.) 11(1/2), 87-110, (2001).
62. Mustafa, A. , Sharif, S., An approach to Adaptive E-Learning Hypermedia System based on Learning Styles (AEHS-LS): Implementation and evaluation. International Journal of Library and Information Science, 3(1), pp. 15-28, (2011)
63. Hanna, M. (2004). Data mining in the e-learning domain. Campus-Wide Inf. Syst. 21, no. 1, pp. 29-34.
64. Merceron, A., K. Yacef. (2005). Educational data mining: A case study. Proc. Int. Conf. Artif. Intell. Educ.
65. Baker, R., K. Yacef. (2009). The state of educational data mining in 2009: A review and future visions." J. Educ. DataMining. Vol 1(1), pp. 3-17
66. Schafer, J. B. (2005) The application of data-mining to recommender szstems. In Encyclopedia of data warehousing and mining, by J. Wang, 44-48. Hersehey: PA. Idea Group,

67. Romero, C., S. Ventura. (2010). Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics- Part C: Applications and Reviews*. Vol 40(6), pp. 601-618.
68. Lazcorreta, E., Botella F., Fernández-Caballero A. (2008). Towards personalized recommendation by two-step modified apriori data mining algorithm. *Expert Systems with Applications*. Vol 35(3). pp. 1422–1429.